

Predikcija proizvodnje energije iz vjetroelektrane strojnim učenjem

Jurić, Marko

Master's thesis / Diplomski rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:168:098514>

Rights / Prava: [In copyright](#)/[Zaštićeno autorskim pravom.](#)

Download date / Datum preuzimanja: **2025-03-29**



Repository / Repozitorij:

[FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repository](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 427

**PREDIKCIJA PROIZVODNJE ENERGIJE IZ
VJETROELEKTRANE STROJNIM UČENJEM**

Marko Jurić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

DIPLOMSKI RAD br. 427

**PREDIKCIJA PROIZVODNJE ENERGIJE IZ
VJETROELEKTRANE STROJNIM UČENJEM**

Marko Jurić

Zagreb, lipanj 2024.

DIPLOMSKI ZADATAK br. 427

Pristupnik: **Marko Jurić (0036525502)**

Studij: Računarstvo

Profil: Znanost o podacima

Mentor: izv. prof. dr. sc. Vinko Lešić

Zadatak: **Predikcija proizvodnje energije iz vjetroelektrane strojnim učenjem**

Opis zadatka:

Vjetar je najzastupljeniji aktualni obnovljivi izvor energije. Proizvodnja energije iz vjetroelektrana promjenjiva je i stohastička varijabla što otežava integraciju velikog broja vjetroatregata u elektroenergetski sustav. U radu je potrebno načiniti algoritam za predikciju proizvodnje energije iz vjetroelektrane koji na temelju povijesnih podataka brzine vjetra, vremenske prognoze i modela vjetroatregata proračunava buduće vrijednosti proizvodnje energije s odabranom vremenskom rezolucijom. Algoritam je potrebno izvoditi u stvarnom vremenu koristeći podatke o vremenskoj prognozi i prošlim vrijednostima brzine vjetra pohranjene u bazu podataka. Predikciju proizvodnje potrebno je konačno vizualizirati u obliku grafova proizvodnje električne energije.

Rok za predaju rada: 28. lipnja 2024.

Iskreno se zahvaljujem mentoru dr. sc. Vinku Lešiću na ukazanom povjerenju, stručnom vodstvu i neizmjernoj podršci tijekom izrade ovoga diplomskog rada. Njegova smjernica i savjeti bili su od ključne važnosti za uspješan završetak rada.

Na kraju, jedno posebno hvala upućujem svojoj obitelji i prijateljima koji su bili uz mene tijekom cijelog studija, pružajući mi neiscrpnu motivaciju i podršku.

Sadržaj

1. Uvod	3
2. Lista potencijalnih izvora podataka	5
3. Vjetroelektrane	8
3.1. Pretvorba energije vjetra	9
4. Strojno učenje	12
4.1. Uvod u strojno učenje	12
4.1.1. primjena	12
4.1.2. Podatci i problemi	13
4.1.3. Osnovni pojmovi	16
4.2. Random forest regresija	19
5. Duboko učenje	21
5.1. Povratne neuronske mreže	21
5.2. Wavelet transformacija	23
6. Predikcija snage vjetroelektrane uz brzinu vjetra	25
6.1. Implementacija	25
6.2. Dataset	26
6.2.1. Modeli i rezultati	28
7. Predikcija brzine vjetra	35
8. Sučelje za prikaz	37

9. Rezultati i rasprava	39
10. Zaključak	41
Literatura	42
Sažetak	44
Abstract	45

1. Uvod

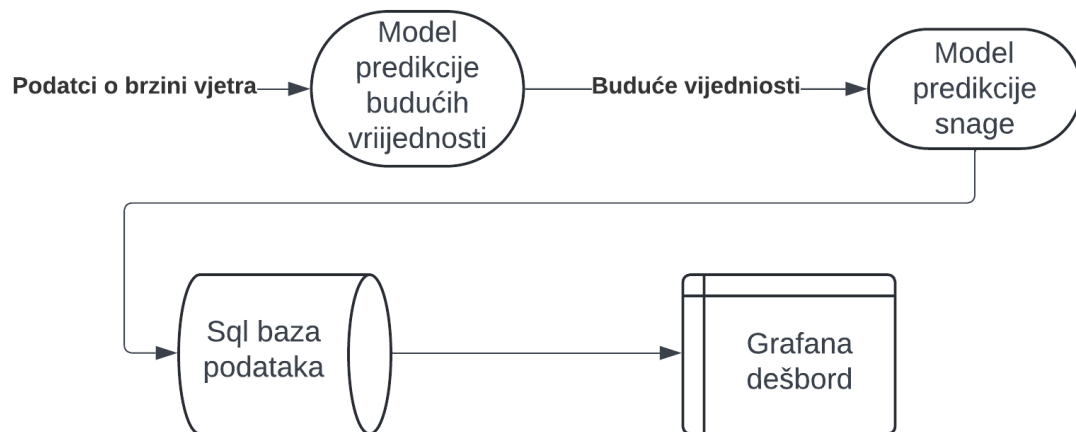
Glavna svrha ovog rada je korištenje strojnog učenja za predikciju proizvodnje energije vjetroelektrane. Strojno učenje, grana umjetne inteligencije, ima sve veću primjenu u različitim industrijama, uključujući energetiku. U kontekstu vjetroelektrana, strojno učenje može igrati ključnu ulogu u optimizaciji proizvodnje energije i predikciji budućih performansi. Kroz analizu povijesnih podataka i vremenskih uvjeta, modeli strojnog učenja mogu precizno predviđati buduće brzine vjetra, što je ključno za efikasno planiranje i operaciju vjetroelektrana.

Sustav koji će se opisati je prikazan na slici 1.1. Prvi korak u sustavu je predikcija buduće brzine vjetra. Ova predikcija temelji se na povijesnim podacima o brzini vjetra. Nakon što se predvidi buduća brzina vjetra, sljedeći korak je predikcija proizvodnje snage vjetroelektrane na temelju dobivenih vrijednosti brzine vjetra. Ovi podaci omogućuju bolje upravljanje resursima i planiranje proizvodnje, čime se povećava učinkovitost i smanjuju operativni troškovi.

Sve predikcije i relevantni podaci šalju se u bazu podataka, gdje se pohranjuju i mogu biti dalje obrađeni i analizirani. Za vizualizaciju podataka koristi se alat Grafana, koji omogućuje pregled i praćenje performansi vjetroelektrane u realnom vremenu. Grafana omogućuje korisnicima da lako razumiju složene podatke putem intuitivnih grafova i dijagrama, što olakšava donošenje odluka temeljenih na podacima.

Uz implementaciju ovog sustava, moguće je značajno poboljšati predikciju proizvodnje energije i optimizirati operacije vjetroelektrana. To dovodi do većeg povrata investicija, smanjenja emisija stakleničkih plinova i doprinosi održivijoj energetskej budućnosti. Korištenje strojnog učenja u ovom kontekstu ne samo da unapređuje tehnologiju vjetroelektrana, već također pomaže u globalnom cilju prelaska na obnovljive izvore

energije i borbe protiv klimatskih promjena.



Slika 1.1. Dijagram sustava

2. Lista potencijalnih izvora podataka

Za potrebe ovog rada, ključno je prikupiti precizne povijesne podatke o vjetru i vremenskim uvjetima s visokom vremenskom i prostornom rezolucijom. Ovi podaci su neophodni u staranju potrebnih modela za predikcije. U ovom poglavlju analizira se niz dostupnih API-ja za prikupljanje meteoroloških podataka, te se procjenjuje njihova prikladnost za ovaj zadatak. Posebna pažnja posvećena je ograničenjima koja mogu utjecati na valjanost i pouzdanost prikupljenih podataka.

- **Weatherstack API**

- **Karakteristike:**

- * Pruža trenutne vremenske podatke i prognoze za naredne dane, kao i povijesne podatke.
 - * Ima veliku pokrivenost cijelog svijeta, ali lošu vremensku i prostornu rezoluciju (1h i 1km).
 - * Besplatan plan ograničen je na 250 poziva mjesečno, dok napredni planovi omogućuju do 50,000 poziva za veće potrebe.

- **Referenca:** [1]

- Neće se koristiti primarno zbog loše vremenske rezolucije.

- **AccuWeather API**

- **Karakteristike:**

- * Pruža detaljne vremenske podatke i dugoročne prognoze.
- * Nema mogućnost pregleda povijesnih podataka.
- * Jako dobra vremenska i prostorna rezolucija(1 minuta, 100metara)
- * Besplatan plan nudi 250 broj upita na dan, dok plaćena verzija omogućuje 225.000 poziva dnevno.

– **Referenca:** [2]

– ne koristimo jer nema povijesne podatke

- **Dark Sky API (sadašnje prognoze samo preko Apple-a)**

– **Karakteristike:**

- * Pruža hiperlokalne i precizne trenutne vremenske podatke i prognoze za naredne sate i dane.
- * Jako dobra ali dostupna samo unutar na apple okruženja.

– **Referenca:** [3]

– Samo dostupna na apple proizvodima stoga ne koristimo

- **ClimaCell API**

– **Karakteristike:**

- * Koristi inovativne tehnologije poput "weather from space" za poboljšanje preciznosti prognoza.

– **Referenca:** [4]

– ne transparentni za oko detalja i cijene

- **WeatherAPI**

– **Karakteristike:**

- * Pruža trenutne vremenske podatke, satne i dnevne prognoze za lokacije širom svijeta.

- * Jako loša vremeska rezolucija od 1h.

- * Besplatan opcija nudi jako puno poziva u danu(1 milijun).

- **Referenca:** [5]

- loša vremeska rezolucija

- **OpenWeatherMap API**

- **Karakteristike:**

- * Pruža trenutne vremenske podatke, kao prognoze od tri sata.

- * Podržava veliki broj lokacija širom svijeta sa jako dobrom vremenskom rezolucijom od 5 min.

- * Besplatan plan nudi milijun poziva u mjesecu što je jako dobro.

- **Referenca:** [6]

- Kasnije korištenje je pokazalo da je vremenska rezolucija 1h ne 5min pa se neće koristiti

Niti jedan api nije dovoljno dobar za korištenje u radu pa je odlučeno koristiti dataset koji će kasnije biti objašnjen.

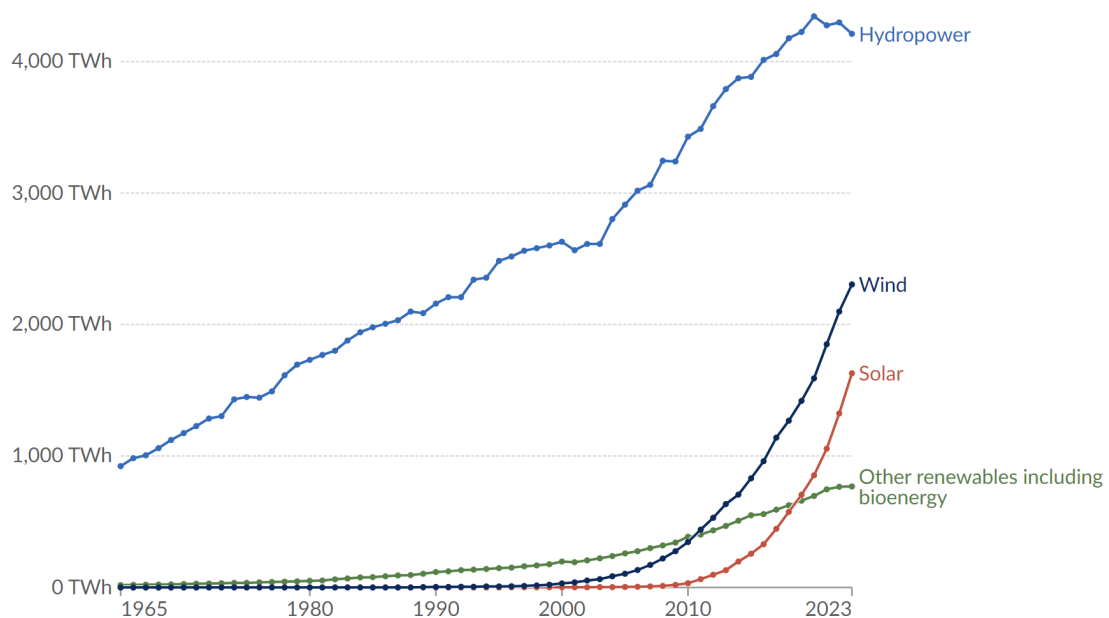
3. Vjetroelektrane

Energija vjetra smatra se glavnim izvorom ekološki održive energije, a buduće generacije će koristiti električnu energiju iz obnovljivih izvora. Sve veća upotreba energije vjetra donijela je obnovljen interes u ovom području. Često razvoj postaja za proizvodnju energije vodi do sigurne i pouzdane održavanja, što jamči ispravnu vidljivost rada. Vjetroresursi su presudni pri odabiru mjesta za izgradnju vjetroagregata, kao i učinkovita i ekonomski poželjna opcija za procjenu potencijala vjetra kako bi se iskoristila investicija i razvoj.

Geopolitička osviještenost, brza industrijalizacija, degradacija okoliša i smanjenje rezervi fosilnih goriva glavni su strateški pokretači istraživanja i razvoja 'zelene energije' kako bi se zadovoljile buduće energetske potrebe. Stoga su dane smjernice za alternativne izvore energije kako bi se borilo protiv energetske krize, zagađenja zraka, visokih emisija i stvorila povoljna atmosfera koja vodi ka zelenijem okolišu. Vlade raznih zemalja planiraju pretvoriti veliki postotak proizvodnje električne energije iz nekonvencionalnih izvora energije, poput hidroenergije, solarne energije, energije vjetra, energije plime i oseke, biomase, male hidroenergije itd., kao i obnovljivih izvora te kombiniranih metoda energije (hibridnih) prema održivom zelenom okolišu. Sektor proizvodnje obnovljive energije postao je dio znanosti o okolišu s interdisciplinarnim pristupom različitim aspektima proizvodnje energije i upravljanja, sigurnim razvojem i obećava čišći okoliš. U ovom tehničkom radu, nalazi iz literature o energiji vjetra, budući globalni status i strategije ublažavanja istraženi su kako bi se razumjeli okolišni izazovi[7]

Godina 2023. bila je prelomna za globalnu industriju vjetra, s rekordnih 117 GW novoinstaliranog kapaciteta. Ovaj nevjerojatan porast od 50% u odnosu na 2022. godinu pokazuje rastuću potražnju i sposobnost industrije da odgovori na globalne energetske izazove. Instalacije su izvedene u 54 zemlje, što potvrđuje široku prihvaćenost i predanost energiji vjetra na svim kontinentima. Uzimajući u obzir trenutni zamah i političku

podršku, GWEC je revidirao svoje prognoze za razdoblje 2024-2030, predviđajući ukupno 1210 GW novog kapaciteta do kraja desetljeća. Ovaj porast odražava sve veći broj nacionalnih politika koje favoriziraju obnovljive izvore energije, kao i povećane investicije u industriju vjetra.[8] Na grafovima 3.1 i 3.2 se vidi brzi rast industrije vjetra pogotovo u državama kao što su Danska i Ujedinjeno Kraljevstvo. Također se može vidjeti jedan problem u jako velikoj iregularnosti u količini proizvodnje pogotovo u Danskoj. Vjetar je po prirodi varijabilan i nepredvidiv, što može dovesti do nestabilne proizvodnje električne energije. Ova neredovitost može uzrokovati poteškoće u održavanju stabilne opskrbe energijom i zahtijeva integraciju s drugim izvorima energije ili sustavima za pohranu energije kako bi se osigurala kontinuitet opskrbe.

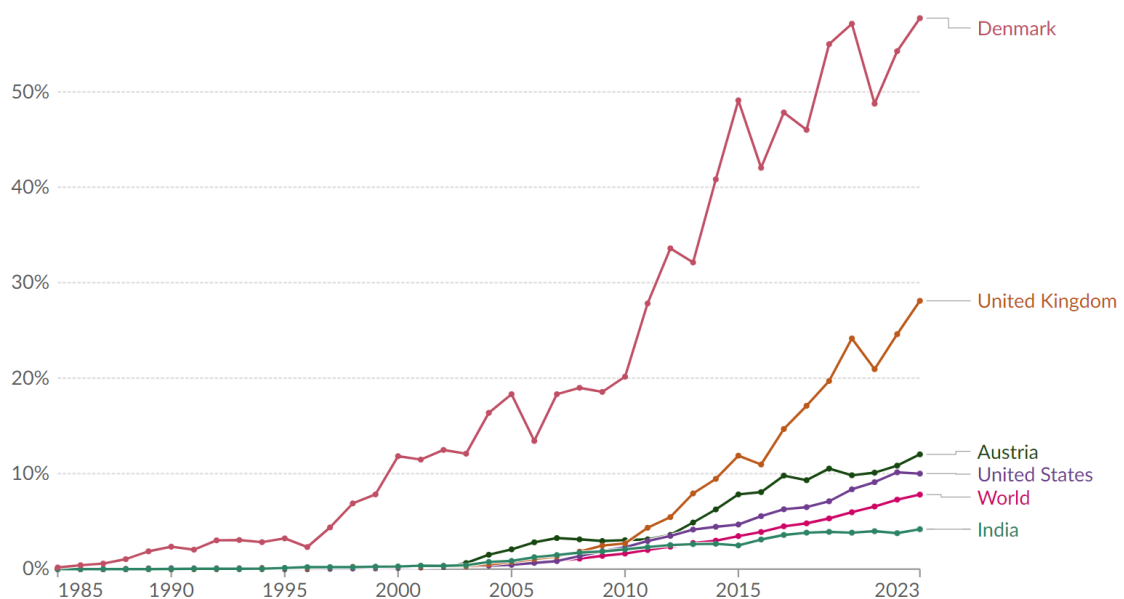


Slika 3.1. Količina energije proizvedena različitim vrstama obnovljivih izvora[9]

3.1. Pretvorba energije vjetra

Vjetroagregati se sastoje od nekoliko ključnih komponenti koje omogućuju pretvaranje energije vjetra u električnu energiju. Glavne komponente uključuju:

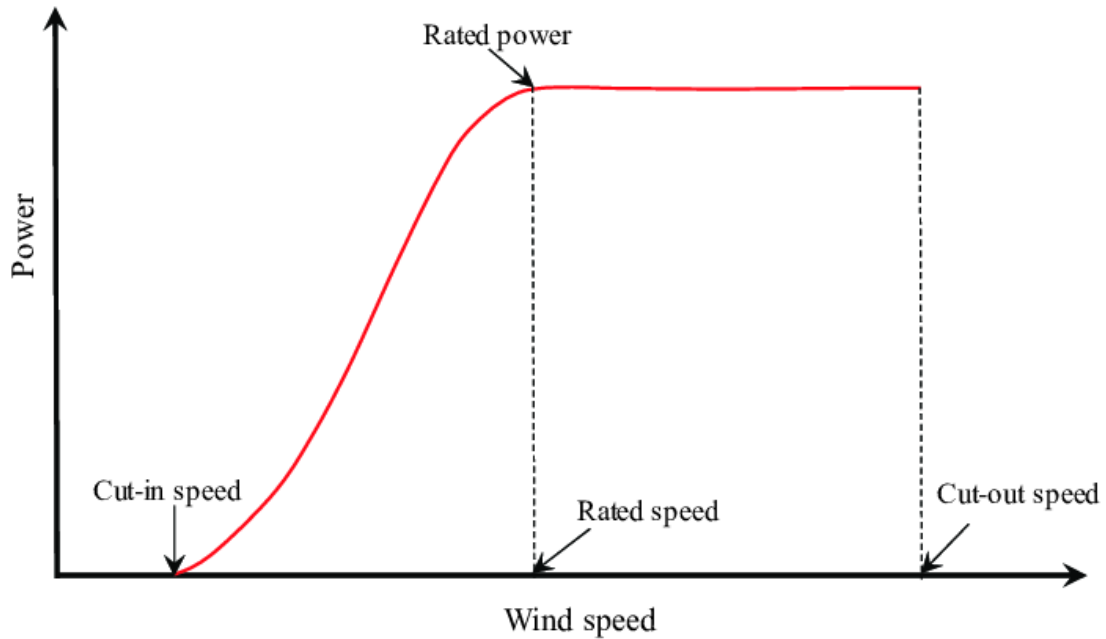
- Lopatice (blades): Pretvaraju kinetičku energiju vjetra u mehaničku energiju. Lopatice su dizajnirane da maksimalno iskoriste dostupnu energiju vjetra kroz aerodinamički profil. Tipično ih ima 3.
- Generator: Pretvara mehaničku energiju rotacije lopatica u električnu energiju.



Slika 3.2. Količina ukupne proizvodnje energije po državi[9]

- Mjenjač (gearbox): Prijenosnik koji povezuje rotaciju lopatica s generatorom, omogućujući optimalne brzine za generiranje električne energije. Mjenjači se koriste za visokobrzinske generatore s manjim radijusom i dužom duljinom. Također postoji modeli koji ne koriste Mjenjač koji imaju istu kutnu brzinu lopatica i generator. Takvi modeli su sve popularniji zbog jednostavnosti održavanja
- Pitch sustav: Mehanizam koji kontrolira kut lopatica kako bi optimizirao njihovu orijentaciju prema smjeru vjetra, omogućujući bolju kontrolu proizvodnje energije i zaštitu sustava od prejakih vjetrova.
- Gondola: Kućište koje sadrži generator, mjenjač i druge komponente, montiranoj na vrhu stupa turbine.

Rad vjetroelektrane podijeljena je u dvije regije ovisnosti o brzini vjetra. Prva je regija vjetra niske brzine koja pokušava uhvatiti što više energije vjetra i regija vjetra visoke brzine gdje je proizvodnja energije konstanta kako bi osigurali dugoročnost vjetroelektrane. Na slici 3.3 Možemo vidjeti 3 ključne brzine vjetra koje utječu na proizvodnju. Prva je cut-in brzina, ispod te brzine vjetra vjetroelektrana mora trošiti više snage na održavanje sustava nego što dobiva od vjetra stoga je gušena. Nakon cut-in brzine i sve do optimalne brzine proizvodnja raste. Nakon optimalne brzine snaga ostaje ista i što se brzina povećava to se lopatice više kreću kako bi koristile kao kočnice. Nakon Cut-out



Slika 3.3. Ovisnost snage o brzini vjetra vjetroelektrane[10]

brzine vjetroelktrana se gasi bog prevelike brzine vjetra. Moderne vjetroelktrane mogu imati ovakvo ponašanje neovisno o smjeru vjetra jer mogu rotirati lopatice kako bi uvijek bile optimalno usmjerene. Koeficijent snage (C_p) može se izraziti kao omjer snage izvučene iz vjetra (P_t) i ukupne dostupne snage vjetra (P_V):

$$C_p = \frac{P_t}{P_V} \quad (2.1)$$

gdje:

$$P_V = \frac{1}{2} \rho_{air} \pi R^2 V^3 \quad (2.2)$$

gdje je (ρ_{air}) gustoća zraka, (R) radijus aerodinamičnog diska i (V) brzina vjetra. Najveća moguća teoretska vijednost je 59.3%, Dok većina modernih je oko 50%. Kombiniranjem ovih formula dobijemo:

$$P_t = C_p(\lambda, \beta) = \frac{1}{2} \rho_{air} \pi R^2 V^3 \quad (2.3)$$

gdje je (λ) brzina vrha lopatice u odnosu na brzinu vjetra i (β) kut nagiba.

4. Strojno učenje

4.1. Uvod u strojno učenje

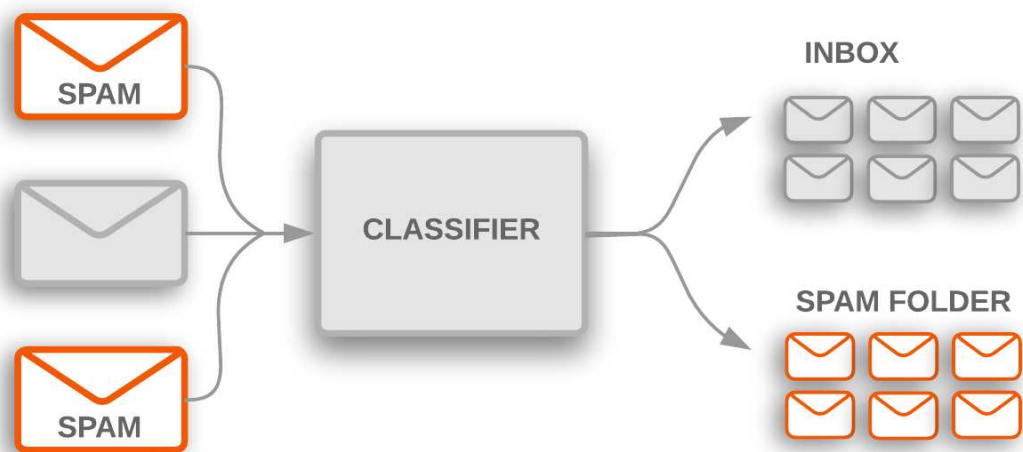
Strojno učenje može se pojaviti u mnogim oblicima. Ovdje ćemo raspravljati o brojnim primjenama, vrstama podataka koje koriste, te formalizirati probleme na stiliziraniji način. To je ključno kako bismo izbjegli ponovno izmišljanje kotača za svaku novu primjenu. Umjesto toga, cilj je smanjiti raspon različitih problema na skup uskih prototipova, a znanost strojnog učenja je riješiti te probleme i pružiti dobra jamstva za rješenja.

4.1.1. primjena

Većina ljudi bit će upoznata s konceptom rangiranja web stranica. To je proces stvaranja upita tražilici, koja zatim pronalazi relevantne web stranice i vraća ih poredane po relevantnosti. Tražilica mora "znati" koje su stranice relevantne, što se može postići analizom strukture poveznica, sadržaja stranica, učestalosti praćenja predloženih poveznica ili pomoću primjera upita s ručno rangiranim stranicama. Sve više se strojno učenje koristi za automatizaciju ovog procesa.

Slična primjena je kolaborativno filtriranje, koje koriste internetske knjižare poput Amazona ili stranice za iznajmljivanje videozapisa poput Netflix. Problem je sličan rangiranju web stranica, ali nedostaje upit. Umjesto toga, koriste se prošle odluke o kupnji i gledanju korisnika kako bi se predvidjele buduće navike. Ključne informacije su odluke sličnih korisnika, zbog čega se proces naziva kolaborativnim. Automatski sustav za rješavanje ovog problema dovodi do bržeg pronalaska željenog proizvoda.

Problem automatskog prevođenja dokumenata također je nejasno definiran. Mogli bismo pokušati potpuno razumjeti tekst prije prevođenja koristeći pravila izrađena od strane stručnjaka za oba jezika, što je vrlo težak zadatak. Umjesto toga, mogli bismo ko-



Slika 4.1. kalsifikacija emaila

ristiti primjere prevedenih dokumenata višejezičnih organizacija (UN, EU) kako bismo naučili prevoditi između jezika. Ovaj pristup strojnog učenja pokazao se uspješnim.

Sigurnosne primjene, poput kontrole pristupa, često koriste prepoznavanje lica. Sustav prepoznaje osobu s fotografije ili video snimke, klasificirajući lice u jednu od kategorija (npr. muškarac, žena) ili odlučujući da je lice nepoznato. Sličan problem je verifikacija, gdje se provjerava je li osoba doista ona za koju se izdaje, što zahtijeva odgovor da/ne. Sustav treba učiti koje su značajke lica relevantne za prepoznavanje u različitim uvjetima osvjetljenja, izraza lica i sl.

Problem klasifikacije često se javlja u praksi i bit će prototip za mnoge primjene u ovoj knjizi. Na primjer, filtriranje neželjene pošte zahtijeva odgovor da/ne na pitanje sadrži li e-mail relevantne informacije. Ovaj problem je ovisan o korisniku: za čestog putnika, e-mailovi o popustima aviokompanija mogu biti korisni, dok za druge mogu biti smetnja. Priroda neželjene pošte mijenja se tijekom vremena, pa sustav treba učiti kako klasificirati nove e-mailove.

4.1.2. Podatci i problemi

Podatci:

- **Vektori:**

- Vektori su najčešći tip podataka u strojnom učenju. Većinom opisuju karakteristike o nekoj stavki na primjer možemo gledati meteorološke vrijednosti nekog mjesta(temperatura, pritisak, količina padalina itd) ili karakteristike ljudi(visina, spol, masa, itd). Jedan od najčešćih problema kod ovakvih podatak jest izabiranje jedinca mjere. Hoćemo li koristiti metre ili kilometre, fahrenheit ili celzsius.

- **Liste:**

- Slične vektorima samo se razlikuju po tome što imaju varijabilni broj karakteristika za svaku stavku. Na primjer neke kuće mogu imati bazen i opis o dimenziji bazena dok neke nemaju. Ovakvi podatci su teži za učiti model

- **Matrice:**

- Matrice se često koriste kada nam treba nekakva informacija o parovima. Na primjer možemo imati gradove u stupcima i redcima dok u matrici piše koliko su udaljeni. Iz te informacije možemo izgraditi kartu pomoću određenih statističkih metoda.

- **Slike:**

- Također su u obliku matrice samo reprezentiraju boju piksela. Često se za analizu provode razne derivacije i transformacije slike kako bi se mogle izvući korisne značajke za efikasno strojno učenje.

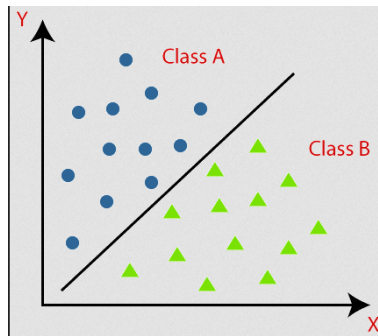
- **Tekst:**

- Tekst je jedna velika kolekcija stringova. Najveći problem kod ovakvih podataka jest konstruirati dobre značajke koje bi bile korisne za učenje. Jedan primjer bag of words koji samo broji koje i koliko puta su se odrađene riječi pojavile. Nakon toga bag of words se može koristiti za klasifikaciju teksta.

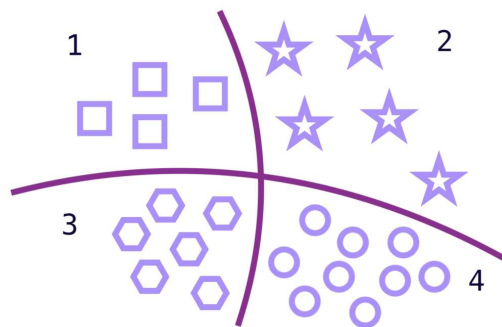
Problemi:

- **Binarna klasifikacija:**

- Ovo je najjednostavniji problem koji se koristio da razvije mnoge naprednije



Slika 4.2. Binarna klasifikacija [11]



Slika 4.3. Multiclass klasifikacija [12]

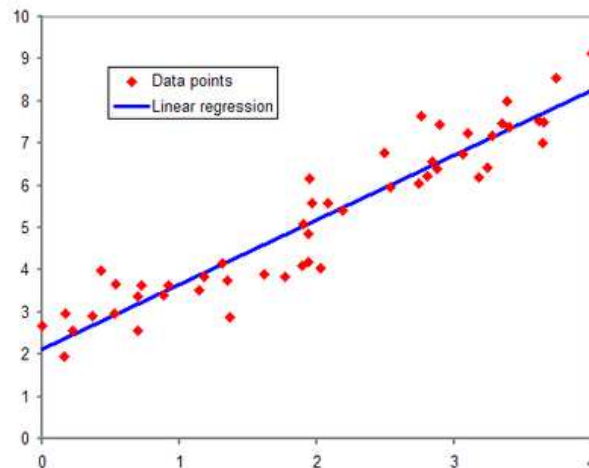
metode koje se sada koriste. Ako imamo podatak x iz domene X želimo pridodati vrijednost $y = \pm 1$. Neki primjeri ovog problema je raspoznavanje da li je na slici pas ili mačka ili ako imao podatke o korisniku banke hoće li uspjeti otplatiti kredit.

- **Muliklasna klasifikacija:**

- Ovo je logičko proširenje binarne klasifikacije samo što y može poprimiti više različitih vrijednosti. Na primjer ako želimo klasificirati neki dokument u kojem je jeziku pisan (Engleski, Hrvatski, Njemački). Jedna bitna činjenica je da nekad želimo veću točnost za određenu klasu naspram drugih. Ako želimo klasificirati stanice kao kancerogene dosta nam je bitno da budemo sigurni da su kancerogene točno klasificirane dok ako klasificiramo zdrave kao kancerogene je manji problem.

- **Regresija:**

- Regresija umjesto da uz podatak x svrsta u y u nekoliko konačnih broj klasa mu pridodaje brojčani vrijednost. Na primjer možemo tražiti vrijednost di-



Slika 4.4. Regrsija[13]

onice sjedeći dan ili kao što ćemo u radu vrijednost brzine vjetra i proizvodnje. Najveća razlika u implementaciji algoritam je stvaranje greške jer sada gledamo i koliko je procjena kriva a ne samo jeli kriva.

4.1.3. Osnovni pojmovi

Kako bi uopće mogli napraviti ikakav model trebaju nam podatci. Većina podataka ima ulazne vrijednosti i izlazne koje treba predvidjeti(Nenadzirano učenje nemaju izlazne). Sljedeći korak je razdvojiti podatke na testni skup i skup za učenje. Razlog tome je ako samo učimo model na jednom skupu podataka model će biti jako dobar na učenim podacima, ali kada mu dođu novi podatci biti će jako loš. Ova pojava se zove prenaučenosť. Nakon što smo podijelili podatke možemo krenuti sa učenjem modela. Postoje tri tipa učenja modela:

- **Nadzirano učenje:**
 - Dobijemo ulaznu varijablu i tražimo izlaznu varijablu(Binarna klasifikacija)
- **Nenadzirano Učenje:**
 - Nemamo zadane izlazne varijable nego želimo otkriti strukturu podataka. Možemo grupirati podatke u clusterne ili im pokušati smanjiti dimenziju. Također vizualizacija podataka jest tip nenadziranog učenja.
- **Podržano učenje:**

- Koriti se slučajevima gdje se informacija o točnosti modela dobije nakon nekoliko odluka kao što su autonomni automobili ili u video igrama.

Učenje se odvija kroz integracije koristeći metodu gradijentnog spusta koji se može opisati ovim formulama. Opća funkcija troška sa L2 regularizacijom je:

$$J(\theta) = \frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)})^2 + \frac{\lambda}{2m} \sum_{j=1}^n \theta_j^2$$

gdje:

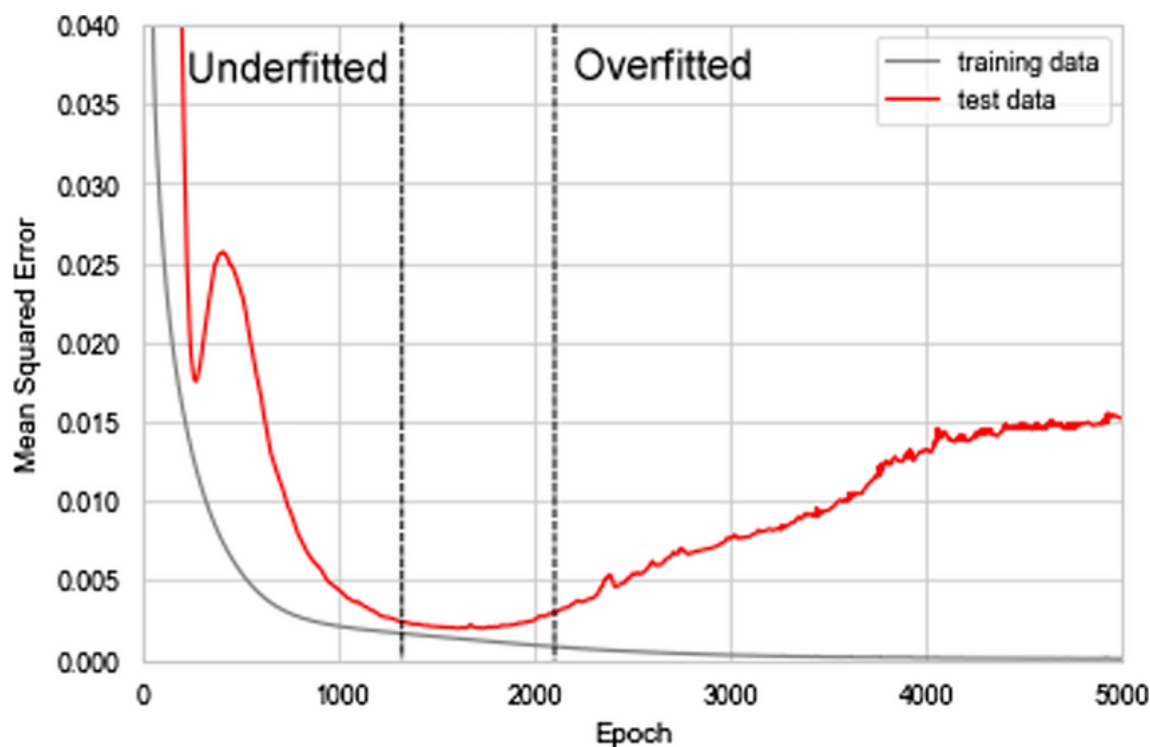
- $h_{\theta}(x)$ je hipoteza ili predikcija modela,
- y su stvarne vrijednosti,
- θ su parametri modela,
- λ je regularizacijski hiperparametar,
- m je broj uzoraka u skupu podataka,
- n je broj značajki (featurea).

Algoritam gradijentnog spusta za ažuriranje parametara θ sa regularizacijom je:

$$\theta_j := \theta_j - \alpha \left(\frac{1}{m} \sum_{i=1}^m (h_{\theta}(x^{(i)}) - y^{(i)}) x_j^{(i)} + \frac{\lambda}{m} \theta_j \right)$$

gdje:

- α je stopa učenja,
- $x_j^{(i)}$ je j -ta značajka i -tog uzorka.
- **Funkcija troška** ($J(\theta)$): Mjera koliko su predikcije modela $h_{\theta}(x)$ blizu stvarnim vrijednostima y . Sastoji se od dva dijela: suma kvadrata grešaka i regularizacijski član koji penalizira velike koeficijente θ .

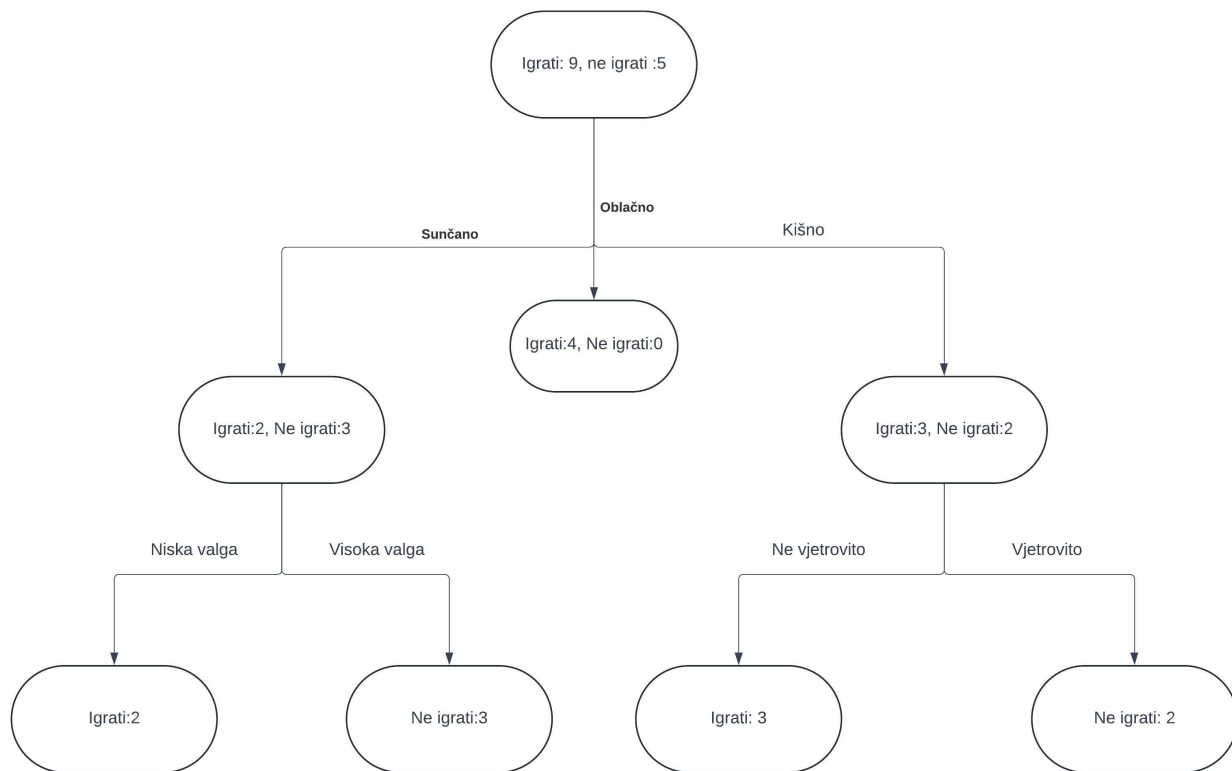


Slika 4.5. Ovisnost greške od broju iteracija za testne i podatke za treniranje

- **Gradijentni spust:** Iterativni optimizacijski algoritam koji koristi gradijent funkcije troška da ažurira parametre θ u smjeru koji smanjuje funkciju troška.
- **Regularizacija:** Dodaje penal funkciji troška kako bi se izbjeglo prekomjerno učenje. L2 regularizacija koristi kvadrat koeficijenata θ .
- **Stopa učenja (α):** Kontrolira veličinu koraka koje algoritam poduzima pri ažuriranju parametara θ .

Ovaj postupak se ponavlja dok se funkcija troška ne konvergira prema minimalnoj vrijednosti ili dok ne dostignemo maksimalan broj iteracija.

Zadnji koncept koji ćemo opisati jest gubitak testno i podataka za treniranje u ovisnosti o broju integracija koje smo proveli u treniranju. Na slici 4.6 možemo vidjeti da se podatci treniranja ponašaju intuitivno, sa iteracijama gubitak pada. Testni podatci s druge strane se ponašaju drugačije na početku greška pada i do tada je model podtreniran. Nakon što dođe do određenog broja iteracije počinje ponovno rasti. To se dogodi jer je model ušao u fazu pretreniranosti. Tada model gubi moć generalizacije i samo može funkcionirati sa podacima za treniranje



Slika 4.6. Primjer algoritma stabla odlučivanja

4.2. Random forest regresija

Sada nam preostaje opisati algoritam koji je dao najbolje rezultate u predikciji snage vjetroelektrane. Random forest algoritam se koristi metodom za učenje koja se zove bagging. Bagging uzima manje pod skupove podataka i trenira jednostavne modele nad njima. Nakon što se modeli iz treniraju ako je u pitanju regresija uzima se prosjek a ako je klasifikacija većinsko glasanje se provodi. Cilj ovakvog algoritma jest smanjiti varijancu konačnog rezultata. Drugi ključni dio random forest regresije su modeli koji se koriste, a to su stabla. Jedan model stabla odluke je jako jednostavan model koji se može jako lako prikazati preko slike 4.7.

Slika prikazuje model koji nam daje odgovor jeli želimo igrati igru tenisa. Imamo dataset od 14 primjera u 9 slučajja se igra, a u 5 ne. Pro pitamo kakvo je nebo i vidimo da ako je oblačno možemo odmah odgovoriti sa da jer su svi primjeri igrati pa ne moramo postavljati dodatna pitanja. Za ostale se varijable problem nastavlja granati ali koncept je isti. Najveća prednost sa stablima odluke je jako ih je lako interpretirati i razumjeti razmišljanje modela. Najveći nedostatak je jako lako se pretreniraju. Sada ćemo spojiti

ova dva koncepta kako bi napravili model sa niskom varijancom i pristranošću. Stabla imaju jako veliku varijancu i korištenje samo veliku količinu neće biti dovoljno, također treba uvesti stupanj nasumičnosti kako bi smanjili korelaciju između njih. Modelu se varijanca ovako ponaša:

$$\rho\sigma^2 + \frac{1-\rho}{B}\sigma^2 \quad (4.1)$$

ρ Je označava korelaciju između stabla, B broj stabala i σ varijancu. Ovdje možemo vidjeti da sa povećanjem broja stabala možemo spustiti varijancu u nula dok prvi dio jednadžbe ostaje isti ali ako eliminiramo korelaciju dobiti ćemo jako malu varijancu. Sada možemo pokazati cijeli algoritam modela:

1. Za $b = 1$ do B :

- (a) Nacrtaj bootstrap uzorak Z^* veličine N iz skupa podataka za treniranje.
- (b) Uzgoji stablo random-forest T_b na bootstrap podacima, rekurzivno ponavljajući sljedeće korake za svaki terminalni čvor stabla, dok se ne dostigne minimalna veličina čvora n_{\min} :
 - i. Odaberi m varijabli nasumično iz p varijabli.
 - ii. Odaberi najbolju varijablu/točku dijeljenja među m .
 - iii. Podijeli čvor na dva kćerina čvora.

2. Izlaz je ansambl stabala $\{T_b\}_{b=1}^B$.

Za predikciju u novoj točki x :

Regresija: $\hat{f}_{\text{rf}}^B(x) = \frac{1}{B} \sum_{b=1}^B T_b(x)$.

Klasifikacija: Neka $\hat{C}_b(x)$ bude klasifikacijska predikcija b -tog random-forest stabla.

Tada $\hat{C}_{\text{rf}}^B(x) = \text{većinski glas}\{\hat{C}_b(x)\}_{b=1}^B$.

5. Duboko učenje

5.1. Povratne neuronske mreže

Duboko učenje predstavlja jednu od najbrže rastućih grana umjetne inteligencije koja omogućuje računalima da uče iz podataka i izvode složene zadatke, poput prepoznavanja objekata na slikama, prevođenja prirodnog jezika, analize zvuka ili čak samostalnog vožnje automobila.

Duboko učenje koristi algoritme temeljene na umjetnim neuronskim mrežama koje su složeni modeli sastavljeni od mnogo manjih jedinica, tzv. neuronskih čvorova. Te neuronske mreže koriste se za prepoznavanje uzoraka u podacima te za donošenje odluka na temelju tih uzoraka.

Umjetne neuronske mreže se sastoje od više slojeva, koji se nazivaju skrivenim slojevima, i koji su sposobni prepoznavati uzorke različitih složenosti. Proces treniranja neuronskih mreža sastoji se od iterativnog prevođenja podataka kroz neuronsku mrežu, pri čemu se parametri mreže prilagođavaju kako bi se minimizirala greška u izlazu.

Duboko učenje se koristi u raznim područjima kao što su računalni vid, obrada prirodnog jezika, preporučiteljski sustavi, robotika i drugi. Također, postoji velik broj biblioteka i okvira za duboko učenje, poput TensorFlow-a, Keras-a, PyTorch-a i drugih, koje olakšavaju izgradnju i treniranje umjetnih neuronskih mreža.

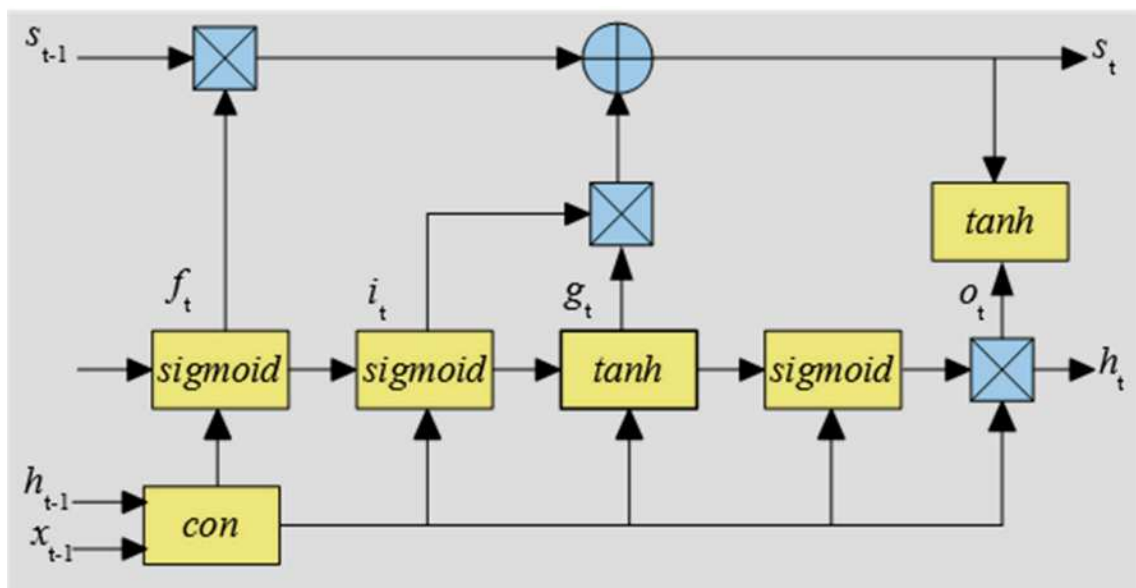
U svakodnevnom životu se duboko učenje koristi u različitim aplikacijama, poput preporuka za kupnju proizvoda, prepoznavanja lica na fotografijama, detekcije oblika na medicinskim slikama, prevođenja teksta i mnogih drugih.

Za naš zadatak potreban nam je model koji može obrađivati podatke slijednoga tipa. Za takav zadatak potrebna nam je mreža koja ne gleda samo svoj input nego i prešli izlaz.

Takve mreže se zovu RNN. Problem takvih mreža je da imaju slabu kontrolu nad gradijentnim spustom i može doći do problema koji zovemo eksplozivajući gradijent ili nestajući gradijent. Eksplozivajući gradijent je problem u gradijentom spustu kada su vrijednosti gradijenta prevelike što dovodi do toga da preskačemo preko minimuma. S druge strane nestajući gradijent problem kada je vrijednost gradijenta premala i ostajemo na istom mjestu. Zbog ovih razloga nam treba sofisticiraniji model.

Model koji ćemo koristiti je LSTM. LSTM mreža ima slojeve koji su posebno dizajnirani za pohranjivanje dugoročne memorije, što omogućava modelu da zadrži informacije o prethodnim stanjima i koristi ih za donošenje odluka o budućim izlazima. To se postiže korištenjem specijalnih "gate" sklopki koje kontroliraju protok informacija kroz mrežu.

Na slici možemo vidjeti da lstm ima 3 ulaza. x predstavlja normalni ulaz modela, h predstavlja prošli izlaz, a s zapamćeno stanje unutar modela. Model se sastoji od 3 dijela. Forget gate, označena s f , računa koliko informacije u s stanju treba zaboraviti. Input gate, označena sa i i g , računa količinu ulaza koju treba zapamtiti. Output gate, označena sa o , računa izlaz.



Slika 5.1. LSTM

$$\begin{aligned}
f_t &= \sigma(W_f [h_{t-1}, x_t] + b_f) \\
i_t &= \sigma(W_i [h_{t-1}, x_t] + b_i) \\
g_t &= \tanh(W_g [h_{t-1}, x_t] + b_g) \\
o_t &= \sigma(W_o [h_{t-1}, x_t] + b_o) \\
s_t &= s_{t-1} \odot f_t + g_t \odot i_t \\
h_t &= \tanh(s_t) \odot o_t
\end{aligned}$$

Iznad su formule za računanje unaprijednog prolaze kroz LSTM. W i b su parametri modela koji se za računanje affine transformacije. Ova modularnost omogućava modelu da nauči određene dijelove modela zasebno. Ta kontrola smanjuje rizik eksplodirajućeg gradijenta i nestajućeg gradijenta. Kako bi ponovno ažurirali parametre računamo parcijalne derivacije gubitka po parametru kojeg želimo ažurirati i množimo sa stopom učenja η kao što možemo vidjeti u formuli ispod.

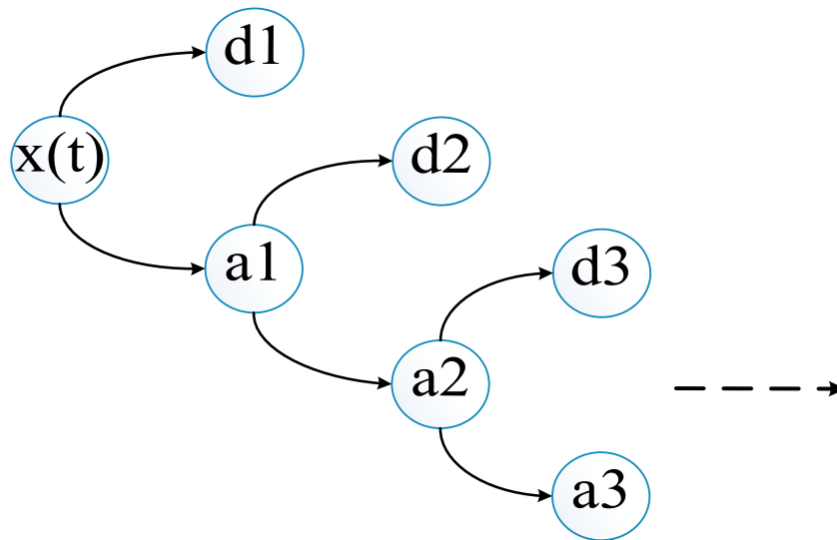
$$\theta_{t+1} = \theta_t - \eta \nabla J(\theta_t) \quad (5.1)$$

5.2. Wavelet transformacija

S obzirom da signal brine vjetra je kompleksan signal na kojeg utječe mnoštvo faktora jedan LSTM model vjerojatno ne bi mogao dobro funkcionirati sa takvim podacima. Kako bi mogli lakše predviđati brzinu vjetra moramo signal dekomponirati na visoke i niske frekvencije koje je lakše koristiti u našim modelima. Mi ćemo koristiti Wavelet transformaciju

Dva su ključna dijela wavelet transformacije:

- **A faktori** predstavljaju aproksimacijske koeficijente, koji opisuju glatke dijelove signala s niskim frekvencijama. Aproksimacijski koeficijenti dobivaju se niskopropusnim filtriranjem signala i uzorkovanjem, što rezultira signalom smanjene frekvencije i razlučivosti.
- **D faktori** predstavljaju detaljne koeficijente, koji opisuju dijelove signala s visokim frekvencijama i detalje signala. Detaljni koeficijenti dobivaju se visokopropusnim



Slika 5.2. Wavelet transformacija

filtriranjem signala i uzorkovanjem, što rezultira signalom s povećanom frekvencijom i detaljima.

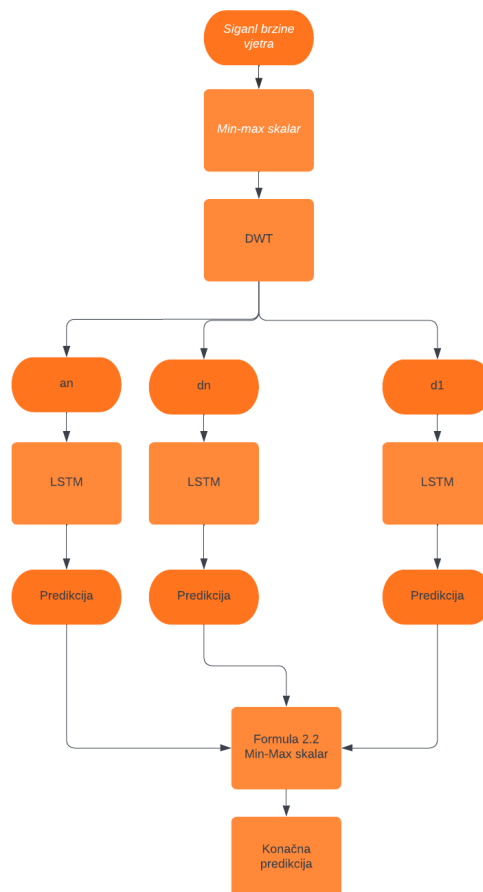
Signal se rekonstruira formulom

$$x(t) = a_n + d_n + d_{n-1} + \dots + d_1 \quad (5.2)$$

6. Predikcija snage vjetroelektrane uz brzinu vjetra

6.1. Implementacija

Generalni nacrt implementacije dan je na slici.



Slika 6.1. Implementacija

Podatke prvo skaliramo. Nakon toga podatci su podijeljeni u 3 skupa, testni skup koji se sastoji od 70% podataka, validacijski skup koji se sastoji 20% podatak i testni skup koji se sastoji 10% podatak. Za treniranje lstm odlučio sam uzeti prošlih 251 intervala od 10 minuta kako bi predvidio sljedećih 5. Kako bi mogao tako trenirati model koristio sam klizni prozor sa korakom 1. Na te podatke je primijenjen wavelet transformacija. Ti podatci se koriste za treniranje lstm modele za svaki dekomponirani signal. Nakon treniranja možemo koristiti validacijske i testne podatke kako bi provjerili točnost modela i radili predikcije u stvarnom vremenu. Hiperparametri modela odabrani su metodom rešetke. Dobiveno je da dubina 1 i 10 neurona u sloju je optimalno rješenje. Model je učen u 30 epoha sa batchevima od veličine 100. Funkcija gubitka je MSE. Za gradijentni spust korišten je ADAM optimizator. Adam je odabran jer je memorijski efikasan i ima promjenjivu stopu učenja.

6.2. Dataset

Skup podataka dolazi iz jedne turbine na kopnenoj vjetroelektrani. Skup podataka pokriva trajanje od jedne godine, ali nedostaju podaci u nekim vremenskim slučajevima. Rezolucija mjerenja je 10 minuta. Za svaku vremensku rezoluciju podaci su raspoređeni u tri stupca. Prvi stupac je vremenska oznaka, drugi stupac je brzina vjetra, a treći stupac je snaga vjetra. Za nedostajuće podatke korištena je interpolacija.

	Time	WindSpeed	Power
0	2014-10-07 01:20:00	8.22	40.526173
1	2014-10-07 01:30:00	8.09	38.552540
2	2014-10-07 01:50:00	7.21	21.313683
3	2014-10-07 02:00:00	6.75	18.031418
4	2014-10-07 02:10:00	6.52	16.593281

Slika 6.2. Dataset

Na slici 6.3 prikazane su osnovne statistike dataseta. Iz prikazanih podataka mogu se uočiti dva ključna aspekta: prvo, prisutnost negativnih vrijednosti za snagu, koje je potrebno zamijeniti s 0; drugo, snaga je izražena u postocima u rasponu od 0 do 100.

Korelacija između varijabli iznosi 0,957, što je izuzetno visok rezultat i u skladu s očekivanjima. Iako korelacija ukazuje na snažnu povezanost, podaci ipak mogu pružiti dodatne uvide.

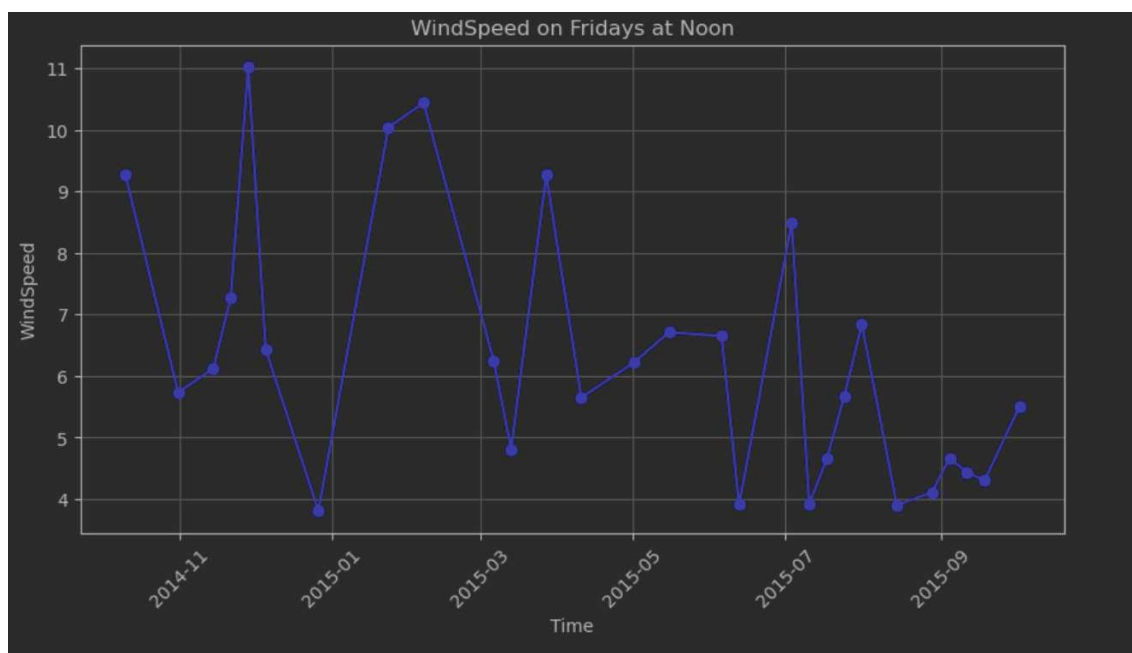
```

Basic statistics:
      WindSpeed      Power
count  39195.000000  39195.000000
mean    6.863497     30.415774
std     2.065032     26.069511
min     0.090000     -0.007123
25%    5.370000     9.727654
50%    6.670000     22.314745
75%    8.070000     44.854827
max    19.790000    100.000000
  
```

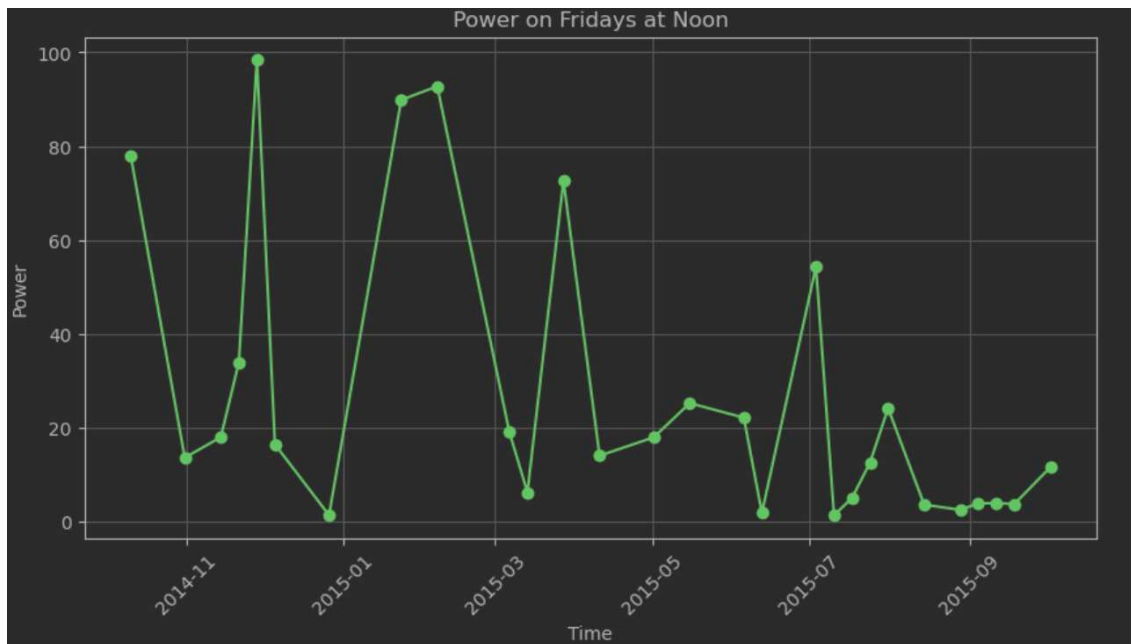
Slika 6.3. Osnovna statistika dataseta

Korelacija	Brzina vjetra	Snaga
Brzina vjetra	1	0.957
Snaga	0.957	1

Na slikama 6.4 i 6.5 Možemo vidjeti brzinu vjetra i proizvodnju snage svaki petak u podne. Na prvi pogled su identični grafovi ali ako pogledamo malo bliže mogu se vidjeti određene razlike koji nam ukazuju na korelaciju od 0.957.



Slika 6.4. Brzina u petak u podne



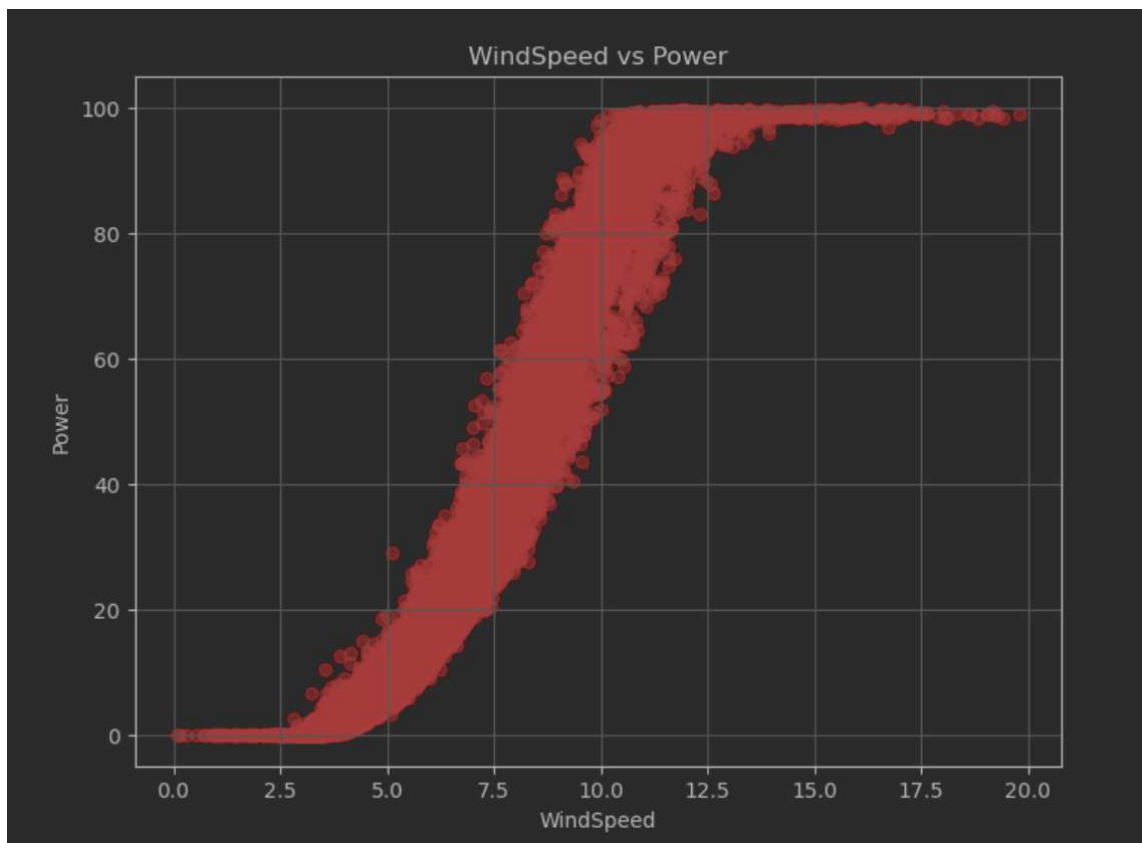
Slika 6.5. Snaga u petak u podne

Kako bih vidjeli tu nepotpunu korelaciju bolje možemo pogledati graf 6.7. Tu vidimo da snaga ne prati idealno brzinu vjetra. Također vidimo da je slična teoretskoj krivulji ovisnosti snage i vjetra. Nakon brzine vjetra od 20 m/s2 snaga je 0 samo se na grafu ne vidi. Ovdje možemo vidjeti da postoji određena varijanca oko teoretske krivulje koja je uzrokovana nekim razlogom. Probati ćemo koristiti prethodne vrijednosti brzine vjetra i gledanjem kada u danu i godini se događa mjerenje kako bi dobili bolju predikciju. Na grafovima 6.6 i 6.7 možemo vidjeti da postoji razlika u variranju od idelane krivulje po noći i danu. Ovo nam može biti korisna informacija u učenju modela kako bi dobili bollje rezultate.

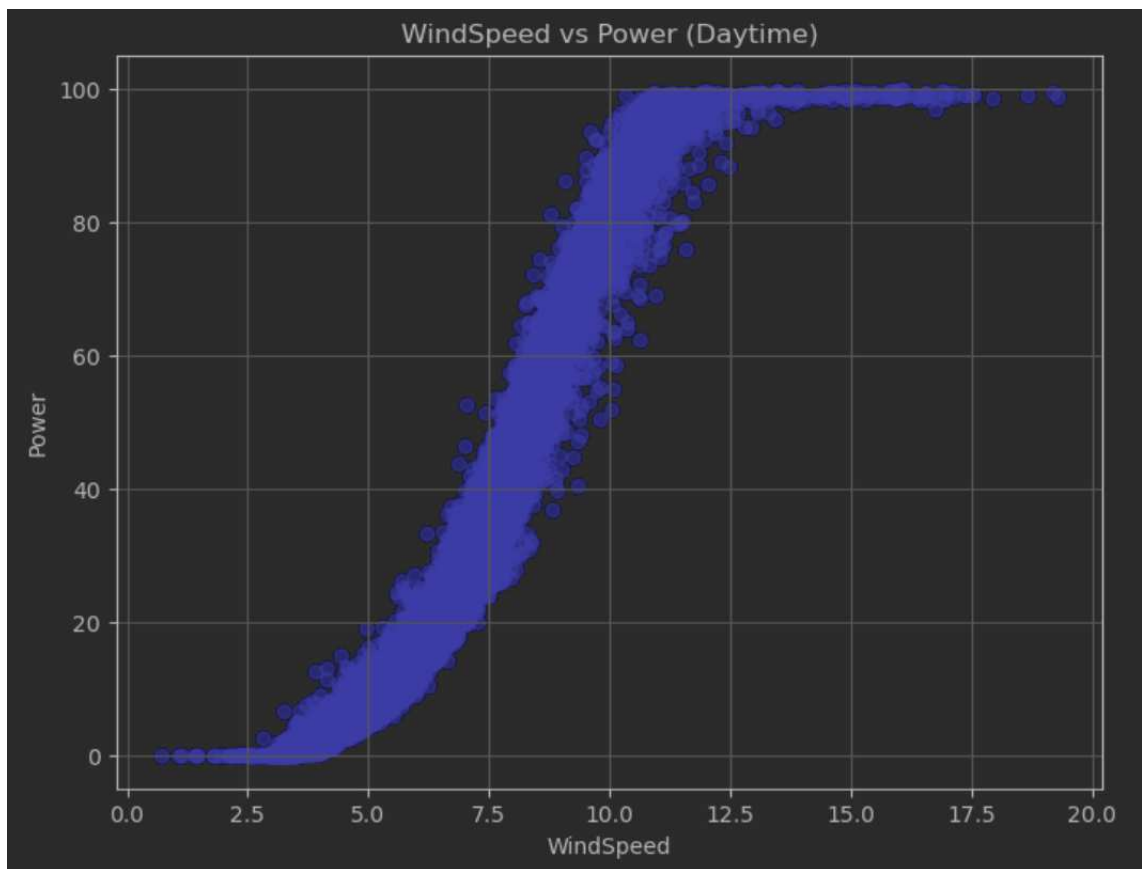
Još ćemo pogledati količini proizvodnje u godini i u danu to se može vidjeti na grafovima 6.9 i 6.8.

6.2.1. Modeli i rezultati

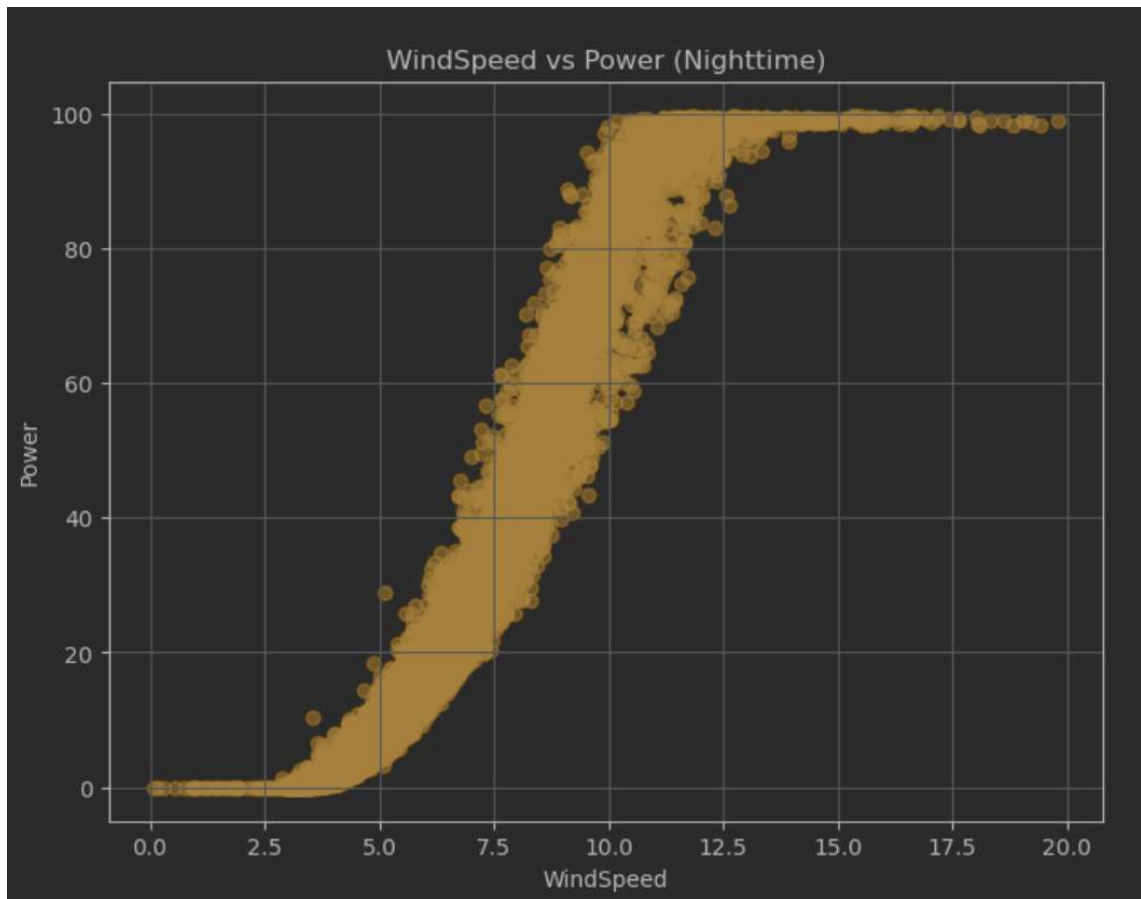
Za predikciju proizvedene snage pokazati ćemo 3 tipa modela. Jedan model je model radnom forest regresije koji se pokazao kao najbolji model od od standardnih modela strojnog učenja. Drugi je LSTM model dubokog učenja. Koristeći duži niz brzina vjetra kako bih bolje predvidio konačni rezultat. Treći model će pokušati samo dobiti idealnu krivulji vjetra i snage. Model će mijenjati 3 varijable i izabrati najbolju kombinaciju tih



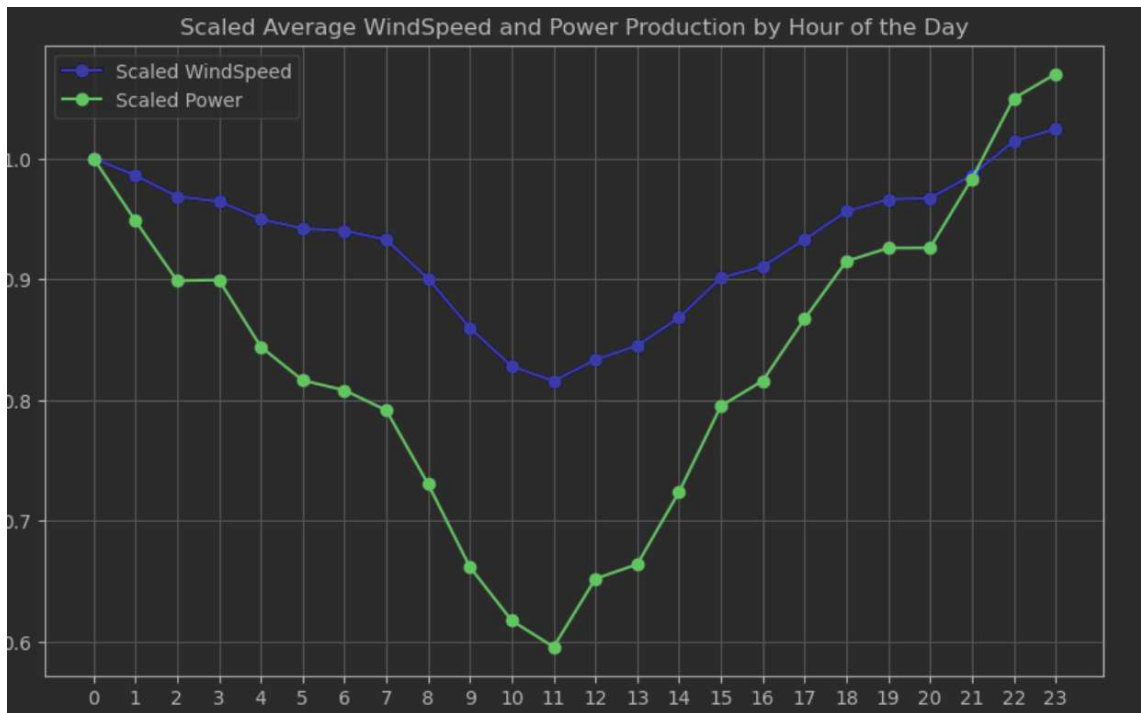
Slika 6.6. Ovisnost brzine vjetra i snage



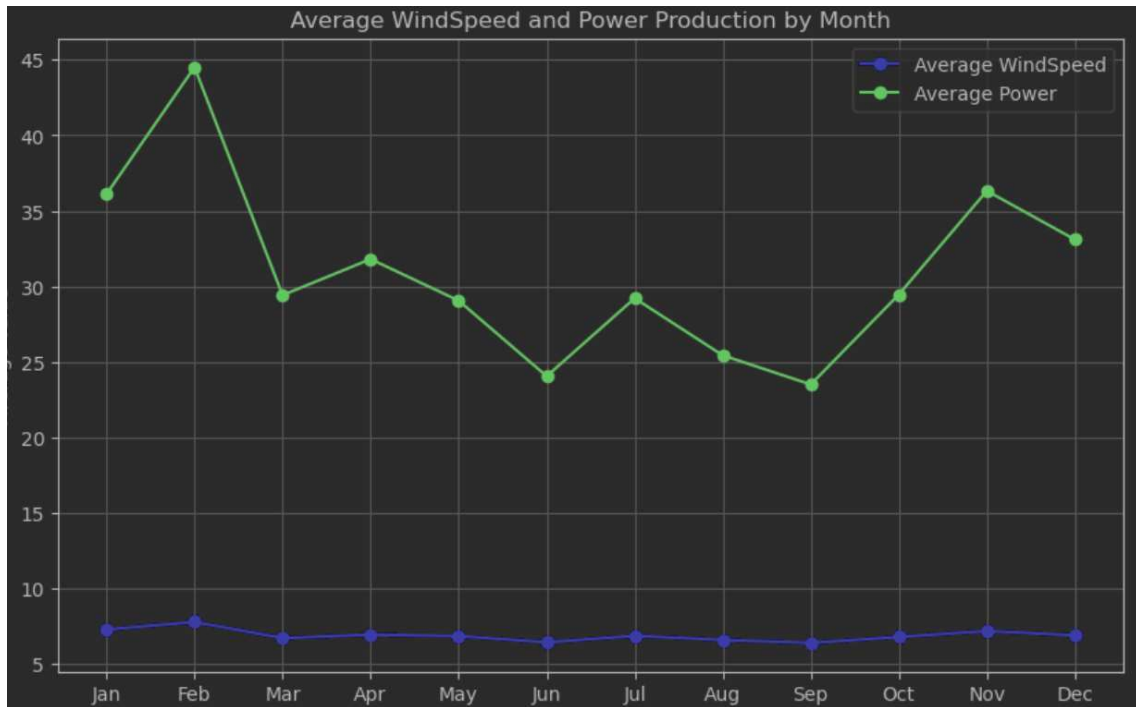
Slika 6.7. Ovisnost brzine vjetra i snage po danu



Slika 6.8. Ovisnost brzine vjetra i snage po noći



Slika 6.9. Prosjek po satu skalirano na prvu vrijdnost



Slika 6.10. Prosjek po mjesecu

varijabli. Prva varijable je donja granica prije koje će model samo predviđati 0 kao rješenje. Druga je gornja granica nakon koje će predikcija uvijek biti 100. One će se pomicati za 0.01 svaku iteraciju. Između te dvije model će napraviti polinomnu regresiju gdje će mijenjati stupnjeve polinoma kao treću varijablu. Ovaj model pokazuje dali ostali modeli uopće korisi jer ako nisu bolji obične regresije onda nisu ni korisni.

Za najbolju forest regresiju korišteni su sljedeći vrijenosti:

- **Dan ili noć**
- **Mjesec**
- **10 prethodnih vrijednosti brzine vjetra:**

Parametri su dobiveni gridsearchom:

- **regressor n estimators:** 100
- **regressor max depth:** none
- **regressor min samples split:** 5

Za najbolju polinomnu regresiju korištena je samo brzina vejtra:

Parametri su dobiveni gridsearchom:

- **prva granica:** 3.05
- **druga granica:** 13.6
- **stupanj polinoma:** 17

Za najbolju lstm regresiju korišteni su samo prošlih 100 vrijednosti brzine vjetra:

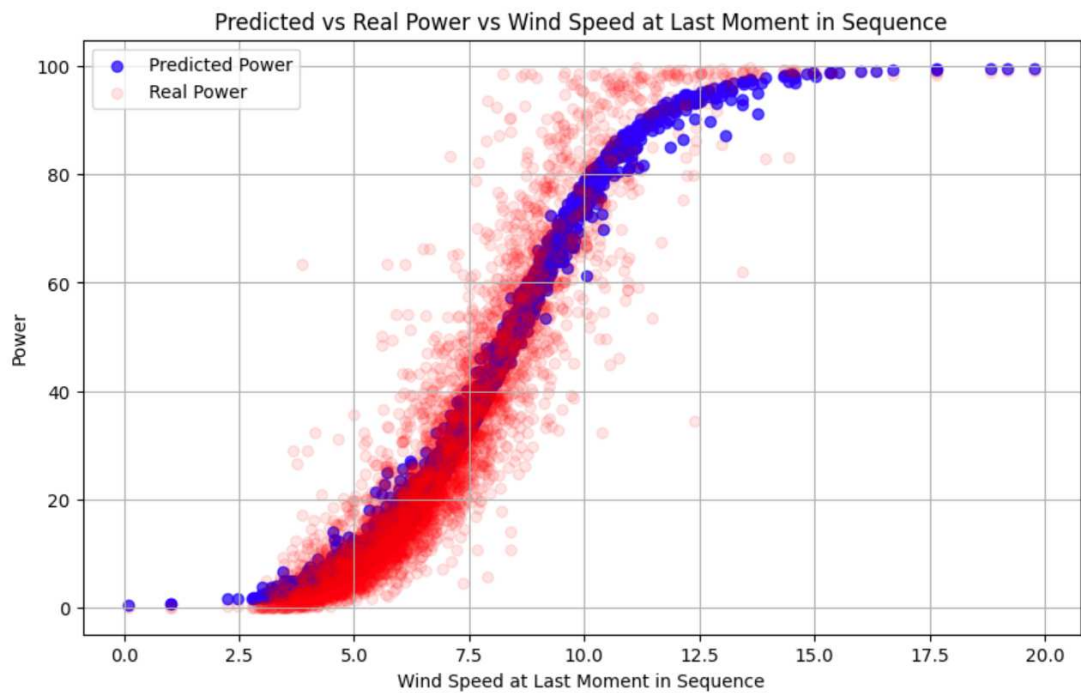
Parametri su dobiveni gridsearchom:

- **ksriveni sloj:** 128
- **Stopa učenja :** 0.01
- **weight decay:** 0.00005

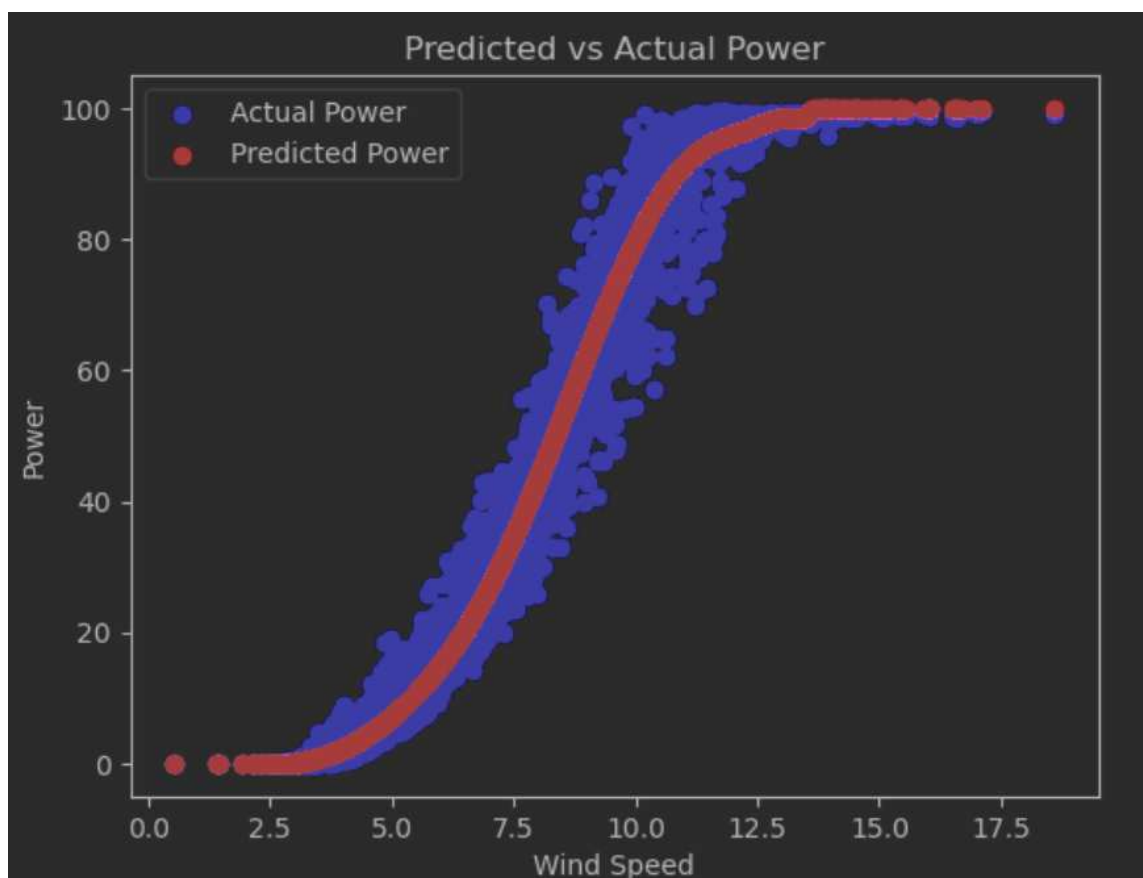
Na grafovima 6.11 - 6.13 možemo vidjeti ova 3 modela. Model koji izgleda da je proizveo podatke najsličnije podacima za treniranje je random forest. Da to potvrdimo moramo usporediti njihovu prosječnu apsolutnu pogrešku.

Model	Random forest	LSTM	Polinomna
MAE	2.3	2.8	2.7

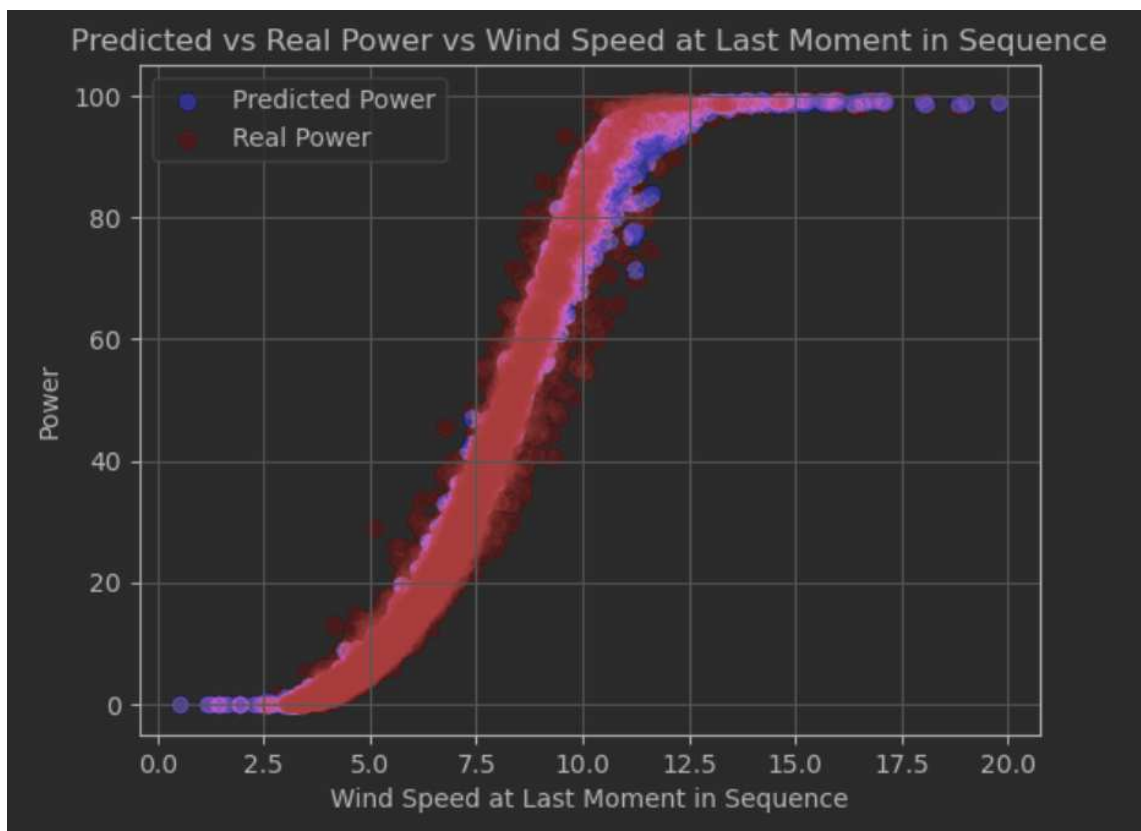
Ovdje možemo vidjeti da je random forest najbolji model i da je uspio biti bolji od obične regresije. To nam pokazuje da uistinu postoji korisne informacije u vrijednostima koje smo koristili u random forestu kako bih dobili bolji rezultat.



Slika 6.11. Lstm regresija



Slika 6.12. polinomna regresija



Slika 6.13. random forest regresija

7. Predikcija brzine vjetra

U ovom poglavlju analiziramo i uspoređujemo tri različita pristupa predikciji brzine vjetra za budućih sat vremena. Prvi model, koji ćemo nazvati naivni model, koristi vrlo jednostavnu strategiju predikcije. Pretpostavlja da će sljedeća vrijednost vremenske serije biti jednaka zadnjoj poznatoj vrijednosti. Ovo je izuzetno jednostavan pristup koji ne zahtijeva nikakve složene algoritme ili obuku modela. Jako je koristan za usporevati sa drugim modelima. Drugi model koristi LSTM mrežu, oblik rekurentne neuralne mreže (RNN) koja je posebno učinkovita za rad sa sekvencijalnim podacima zbog svoje sposobnosti zadržavanja dugoročnih zavisnosti. Treći model kombinira više LSTM mreža koje uče na wavelet s 4 stupnja razlaganja transformacijama vremenskih serija. Wavelet transformacija razlaže vremensku seriju na različite frekvencijske komponente, omogućavajući LSTM mrežama da uče na različitim razinama detalja. Parametri lstm mreže su:

- **ksriveni sloj:** 250
- **Stopa učenja :** 0.01
- **weight decay:** 0

Na ovoj tablici su MAE modela u m/s

MAE	Naivni	LSTM	LSTM-wavelet
10min	0.409	0.791	0.320
20min	0.707	0.988	0.673
30min	0.964	1.225	0.794
40min	1.227	1.384	0.986
50min	1.447	1.502	1.105

Na rezultatima se može vidjeti da j obični lstm gori od od naivnog modela dok je lsmtm

koji koristi wavelet transformaciju značajno bolji u predikciji brzine vjetra.

8. Sučelje za prikaz

Aplikacija za praćenje proizvodnje vjetroelektrane koristi Flask, MySQL, API i Grafanu kako bi omogućila prikupljanje, pohranu, obradu i vizualizaciju podataka o proizvodnji električne energije. Flask je zadužen za backend dio aplikacije. Flask aplikacija šalje upite api svakih par minuta kako bih dobila informacije o brzini vjeta i izračunava proizvedenu snagu. MySQL baza služi za pohranu tih podataka, omogućujući njihovo učinkovito pretraživanje i organizaciju. Grafana finalno koristi podatke u bazi kako bih prikazala samoažurirajuće podatke u lijepom formatu.



Slika 8.1. Grafana dashboard

Iako je aplikacija za praćenje proizvodnje vjetroelektrane uspješno implementirana korištenjem Flask-a, MySQL-a, API-ja i Grafane za prikupljanje, pohranu i vizualizaciju podataka, nisam uspio povezati aplikaciju s modelima za predikciju budućih vrijednosti. Unatoč pokušajima integracije modela za prediktivnu analizu koji bi omogućili prognoziranje buduće proizvodnje električne energije na temelju povijesnih podataka, došlo je do tehničkih poteškoća. Ovi problemi su uključivali neusklađenost između strukture podataka i zahtjeva modela, kao i izazove u osiguravanju točnosti i pouzdanosti predikcija. Kao rezultat toga, funkcionalnost predviđanja trenutno nije dostupna u aplikaciji, ali se planira dodatni rad na integraciji kako bi se ta značajka omogućila u budućnosti.

Osim postojećih vizualizacija u Grafani koje omogućuju praćenje trenutne proizvodnje energije, povijesnih trendova i meteoroloških uvjeta, na *dashboardu* se mogu dodatno implementirati sljedeće vizualizacije kako bi se poboljšala funkcionalnost i korisničko iskustvo:

- **Vizualizacija korelacije** između vremenskih uvjeta (kao što su brzina vjetra, temperatura i atmosferski tlak) i proizvodnje električne energije. Ova analiza može pomoći u identifikaciji ključnih faktora koji utječu na proizvodnju.
- **Heatmap prikazi** za vizualizaciju vremenske distribucije proizvodnje energije kroz različite vremenske periode (npr. po satima tijekom dana, danima u tjednu ili mjesecima u godini). Ovo može omogućiti identificiranje uzoraka u proizvodnji energije.
- **Geografske vizualizacije** koristeći kartografske prikaze za prikaz raspodjele proizvodnje energije na različitim lokacijama u slučaju da sustav pokriva više vjetroelektrana na različitim geografskim područjima.
- **Scenariji simulacija** koji omogućuju korisnicima da unesu različite pretpostavke o budućim uvjetima (npr. promjene u cijenama energije, uvođenje novih tehnologija) i vide simulirane rezultate na proizvodnju energije i financijske pokazatelje.

Implementacija ovih dodatnih vizualizacija može znatno unaprijediti korisničko iskustvo i omogućiti bolji uvid u performanse vjetroelektrane, kao i donijeti dodatnu vrijednost korisnicima kroz analizu podataka i donošenje odluka.

9. Rezultati i rasprava

Rezultati ovog istraživanja pokazali su značajnu prednost korištenja modela strojnog učenja za predikciju proizvodnje energije vjetroelektrana u usporedbi s pristupom bez predikcije. Modeli strojnog učenja su omogućili preciznu predikciju budućih brzina vjetra, što je ključno za optimizaciju proizvodnje energije. Korištenjem povijesnih podataka i meteoroloških informacija, modeli su uspješno predvidjeli promjene u brzini vjetra, čime je omogućeno bolje planiranje i upravljanje vjetroelektranama.

Predikcije proizvodnje energije na temelju brzine vjetra pokazale su se korisnima u povećanju efikasnosti i smanjenju operativnih troškova. Usporedba rezultata modela strojnog učenja s tradicionalnim metodama bez predikcije jasno je pokazala superiornost prvog pristupa. Grafovi i dijagrami generirani putem alata Grafana omogućili su vizualizaciju podataka u realnom vremenu, olakšavajući donošenje informiranih odluka.

Međutim, unatoč pozitivnim rezultatima, tijekom istraživanja su identificirani određeni nedostaci. Nedostatak vremena i resursa ograničio je razvoj složenijih i preciznijih modela strojnog učenja. Napredniji modeli, koji bi mogli uključivati dublje neuronske mreže ili sofisticiranije metode obrade podataka, vjerojatno bi dali još preciznije predikcije i bolju optimizaciju proizvodnje energije.

Jedan od ključnih problema s kojim smo se suočili bila je nemogućnost povezivanja baze podataka s modelima strojnog učenja. Iako su predikcije bile precizne, nedostatak integracije s bazom podataka otežao je automatsku pohranu i daljnju analizu podataka. Ovaj tehnički problem spriječio je potpuno iskorištavanje potencijala sustava i zahtijeva dodatni rad kako bi se osigurala potpuna funkcionalnost.

Unatoč ovim izazovima, rezultati ovog istraživanja jasno pokazuju potencijal i korist korištenja strojnog učenja u predikciji proizvodnje energije vjetroelektrana. Daljnjim ra-

zvojem i rješavanjem tehničkih problema, očekuje se da će se postići još veća učinkovitost i pouzdanost u radu vjetroelektrana, što će doprinijeti održivijoj i ekološki prihvatljivijoj proizvodnji energije.

10. Zaključak

U ovom radu istraženo je korištenje strojnog učenja za predikciju proizvodnje energije vjetroelektrana. Strojno učenje, kao napredna grana umjetne inteligencije, pokazalo se kao moćan alat u optimizaciji proizvodnje energije i predikciji budućih performansi u industriji energetike. Primjena modela strojnog učenja omogućila je preciznu analizu povijesnih podataka i vremenskih uvjeta, što je rezultiralo točnim predviđanjima budućih brzina vjetra.

Razvijeni sustav, koji uključuje predikciju brzine vjetra i proizvodnje snage, pokazao se učinkovitim u poboljšanju operacija vjetroelektrana. Integracijom predikcija i stvarnih podataka u bazu, te korištenjem alata Grafana za vizualizaciju, postignuta je bolja upravljivost i preglednost nad proizvodnjom energije. Ova metoda omogućuje operativcima vjetroelektrana donošenje informiranih odluka, optimizaciju resursa i smanjenje operativnih troškova.

Korištenjem strojnog učenja u ovom kontekstu doprinosi se ne samo učinkovitosti i ekonomičnosti vjetroelektrana, već i globalnom cilju prelaska na obnovljive izvore energije. Preciznije predikcije i optimizacija proizvodnje direktno utječu na smanjenje emisija stakleničkih plinova, što je ključno u borbi protiv klimatskih promjena. Ovaj rad stoga potvrđuje važnost i potencijal strojnog učenja u energetici te postavlja temelje za daljnja istraživanja i napredak u ovom području.

Literatura

- [1] Weatherstack, “Weatherstack api”, <https://weatherstack.com/>, accessed: 2024-09-03.
- [2] AccuWeather, “Accuweather api”, <https://developer.accuweather.com/>, accessed: 2024-09-03.
- [3] A. Inc., “Weatherkit documentation”, <https://developer.apple.com/documentation/weather>, accessed: 2024-09-03.
- [4] C. (Tomorrow.io), “Climacell (tomorrow.io) weather api”, <https://www.climacell.co/weather-api/>, accessed: 2024-09-03.
- [5] WeatherAPI.com, “Weatherapi”, <https://www.weatherapi.com>, accessed: 2024-09-03.
- [6] OpenWeatherMap, “Openweathermap api”, <https://openweathermap.org/api>, accessed: 2024-09-03.
- [7] D. V. Boguslavsky, K. S. Sharov, i N. P. Sharova, “Using alternative sources of energy for decarbonization: A piece of cake, but how to cook this cake?” 2022. [Mrežno]. Adresa: <https://www.ncbi.nlm.nih.gov>
- [8] Global Wind Energy Council, “Global Wind Report 2024”, Online, 2024., [Mrežno]. Adresa: <https://gwec.net/global-wind-report-2024/>.
- [9] H. Ritchie i M. Roser, “Renewable energy”, <https://ourworldindata.org/renewable-energy>, 2020., accessed: 2024-09-03.

- [10] ResearchGate, “Typical wind power output versus wind speed”, <https://www.researchgate.net/figure/Typical-wind-power-output-versus-wind-speedi1323434457>, accessed: 2024-09-03.
- [11] B. Ghavimi, “Binary classification made simple with tensorflow”, <https://www.freecodecamp.org/news/binary-classification-made-simple-with-tensorflow>, 2020., accessed: 2024-09-03.
- [12] W. Commons, “File: Multiclass classification”, https://commons.wikimedia.org/wiki/File:Multiclass_classification.png, accessed: 2024-09-03.
- [13] Wikipedia, “Regression analysis”, https://en.wikipedia.org/wiki/Regression_analysis, accessed: 2024-09-03.
- [14] T. Jonsson, P. Pinson, H. A. Nielsen, H. Madsen, i T. S. Nielsen, “Forecasting electricity spot prices accounting for wind power predictions”, *IEEE Transactions on Sustainable Energy*, sv. 4, br. 1, str. 210–218, 2013. <https://doi.org/10.1109/TSTE.2012.2212731>
- [15] G. Sideratos i N. D. Hatziargyriou, “Probabilistic wind power forecasting using radial basis function neural networks”, *IEEE Transactions on Power Systems*, sv. 27, br. 4, str. 1788–1796, 2012. <https://doi.org/10.1109/TPWRS.2012.2187803>
- [16] C. Potter i M. Negnevitsky, “Very short-term wind forecasting for tasmanian power generation”, *IEEE Transactions on Power Systems*, sv. 21, br. 2, str. 965–972, 2006. <https://doi.org/10.1109/TPWRS.2006.873421>
- [17] J. Lee, W. Wang, F. Harrou, i Y. Sun, “Wind power prediction using ensemble learning-based models”, *IEEE Access*, sv. 8, str. 61 517–61 527, 2020. <https://doi.org/10.1109/ACCESS.2020.2983234>
- [18] A. Kusiak, Z. Song, i H. Zheng, “Anticipatory control of wind turbines with data-driven predictive models”, *IEEE Transactions on Energy Conversion*, sv. 24, br. 3, str. 766–774, 2009. <https://doi.org/10.1109/TEC.2009.2025320>

Sažetak

Predikcija proizvodnje energije iz vjetroelektrane strojnim učenjem

Marko Jurić

Ovaj diplomski rad predstavlja razvoj sustava za predviđanje proizvodnje vjetra koristeći strojno učenje, s vizualizacijom rezultata na Grafana dashboardu. Predviđanje proizvodnje vjetra je ključno za optimizaciju rada vjetroelektrana i integraciju obnovljivih izvora energije u elektroenergetski sustav. Cilj ovog rada je kreirati precizan model koji može predvidjeti količinu proizvedene energije iz vjetra na temelju povijesnih podataka i meteoroloških prediktora. Uspoređeni su Random forest, polinomna regresija i long short-term memory povratna neuronska mreža pristupi od kojih najbolje rezultate pokazuje pristup long short-term memory povratna neuronska mreža uz srednju apsolutnu pogrešku predikcije od 0.32 m/s za 10 min unaprijed do 1.105 m/s za 50 min unaprijed.

Ključne riječi: strojno učenje; grafana; vjetroelektrane; duboko učenje

Abstract

Predikcija proizvodnje energije iz vjetroelektrane strojnim učenjem

Marko Jurić

This thesis presents the development of a system for predicting wind energy production using machine learning, with visualization of results on a Grafana dashboard. Predicting wind energy production is crucial for optimizing the operation of wind farms and integrating renewable energy sources into the power grid. The goal of this work is to create an accurate model that can predict the amount of energy produced from wind based on historical data and meteorological predictors. The approaches compared were Random Forest, polynomial regression, and long short-term memory recurrent neural networks, with the long short-term memory recurrent neural networks approach showing the best results, achieving a mean absolute prediction error ranging from 0.32 m/s for 10 minutes ahead to 1.105 m/s for 50 minutes ahead.

Keywords: machine learning; grafana; windturbine; deep learning